BACKGROUND OF THE INVENTION

1.Field of the Invention

A system relates to the up-to-date of firmware for a computer system is disclosed, more particularly, the purpose firmware can be safely replaced instead of the original firmware.

2.Description of the Prior Art

The recent technology for upgrading firmware is employed by using the combination application of computer software and firmware. Normally, the purpose firmware data of the computer system will be read whereby software operation and data transformation to the related system through transformation media.

Thus, the flow chart of computer system operation according to the prior art is shown as Figure 1. When the method starts, the prior firmware 10 will be removed as 11. Then the new firmware is set up into computer memory as 12. Thus the original firmware can be removed from the saved memory space of computer system using the above method. Consequentially the new firmware can be written into the memory space quickly.

However, if the computer system is suddenly stopped operating without the electrical power support, the computer system will be possibly destroyed. Especially, no doubt, the sudden breakdown to a computer system at the time in the original firmware being upgraded, the computer system will be totally smashed by the above accident.

SUMMARY OF THE INVENTION

In accordance with the present invention, a method is

provided for protecting and upgrading the firmware that substantially solves the above mentioned system error and possible faults to computer system. It certainly can be employed to necessity of the scanner as well. Therefore, the method of this present invention will be described as the below statement. The computer memory for saving firmware firstly will be divided as five portions. The five portions also can be indicated as the following.

The first portion is defined as an initialized program. The function of first portion is for protecting recent firmware situation and processing the movement of prior-edition firmware and then installing new-edition firmware into computer memory. It can be guaranteed that this initialized program never been not modified when movement and installing is under operation. Then, the second portion is a real firmware for controlling computer systems, which provides a new-edition firmware loading. The third portion is a backup firmware of computer system. This portion is always correct and will be executed. Even though a new-edition firmware is failed to install into the computer memory, the third portion still can repair the effort of third portion of firmware. Finally, the fourth portion and the fifth portion can save the parameter of the second and the third portion of firmware, such as the volume of computer files and the value of checksum.

When the firmware is under upgraded process, the second portion of firmware can be successfully updated if there is no error happened. The initialized program will backup the second portion into the third portion of firmware. When the errors happen under upgraded process, the initialized program will write back the backup file of the third portion to the second portion. Therefore, the computer system under operation still can recover the original firmware of original system, even though the errors happen under upgraded process. It can guarantee the system is under a normal condition. Also, it can set up the firmware under the safety condition. Of course, the parameter of all new five portions will be

successfully renewed.

BRIEF DESCRIPTION OF THE DRAWINGS

Further details of the present invention will be apparent to the those who skilled in the art by reference to the exemplary embodiment in the drawing in which:

Figure1 illustrates the flow chart of the conventional technique according to the prior art;

Figure 2 illustrates the flow chart of this method according to the present invention; and

Figure 3 illustrates the relationship between the five conditions according to the present invention.

Table 1 illustrates the four portions of the memory of computer system according to this present embodiment.

DESCRIPTION OF THE PREFERRED EMBODIMENT

The method of the present invention is applied to a broad range of firmware related range and can be from a variety of related-invention. The following description discusses several presently preferred embodiments of the method of the present invention as implemented in process, since the majority of currently available is fabricated in foundry and the most commonly encountered applications of the present invention will involve problems from trial and errors method. Nevertheless, the present invention may also be advantageously employed in any sort of computer technology. Accordingly, application of the present invention is not only intended to be limited to those devices fabricated in silicon semiconductor materials, but also will include those fabricated in one or more of the available.

Thus, the following is a description of the present invention.

The invention will firstly be described with reference to one exemplary structure. Some variations will be described as well as advantages of the present invention. A preferred method of fabrication will then be discussed. It can be that the disclosure of a digital computer having a memory containing a program which is executed by the computer.

The preferred embodiment of this present invention will be described as the following. As Table 1, firstly, the computer memory will be divided as the following five portions. The first portion as Initial Program is not modified but fixed up. Also, this Initial Program will be executed when computer system starts.

Table 1:

Initial Program
Firmware (P1)
Backup Firmware (P2)
P1 Firmware Parameter
P2 Firmware Parameter

Then checksum value of first portion of this firmware is calculated and is compared with the parameter of fourth portion for checking if it is proper. Then, the second portion is Firmware P1, the third portion is Backup Firmware P2. The fourth portion will be P1 Firmware Parameter according to Table 1 and the fifth portion will be P2 Firmware Parameter according to Table 1. Figure 2 also shows the inter-relationship between the following four conditions.

Normally, referring with Figure 2, the first portion of this computer firmware is defined as an Initial Program so that this program will not be modified but will be fixed. Also, this first portion of firmware will be executed while the system starts. Then, the second portion of Firmware (P1) is read and is calculated. Thus,

checksum value of Firmware P1 will be obtained. The above checksum value of Firmware P1 can be compared with the parameter of the fourth portion and can be check if it is correct. When the result is correct, the computer system can be continued.

Sequentially, as Figure 2, the third portion of Firmware (P2) is read and is calculated. Thus, checksum value of Firmware P2 will be obtained. The above checksum value of Firmware P2 can be compared with the parameter of the fifth portion and can be check if it is correct. When the result is correct, the computer system can be continued.

Thus, the third portion of Firmware P2 can be checked. Here, Firmware P1 will be error at the last operation for refreshing the firmware of computer system if Firmware P1 is incorrect. At this time, Initial Program should write back to Backup Firmware P2 to Firmware P1 of the second portion. Then, the original computer program can be repaired so that the computer system can still run the correct operation.

It is shown as Figure 2, after checking Firmware P1, the Backup Firmware P2 will be sequentially checked. Firstly, Backup Firmware P2 can be read and calculated its checksum value by Initial Program. Then the result can be compared with the parameter of the fourth portion and fifth portion of firmware, so that the next checking can be executed if Backup Firmware P2 is correct. However, if Backup Firmware P2 is incorrect, it can be ensured that there are errors of Backup Firmware P2 happened at the last duplicating firmware process. Therefore, at the same time, Initial Program should write Firmware P1 of computer system into Backup Firmware P2 of computer system and should restart the computer system again in order to execute Backup Firmware P2 of computer system. It can keep Backup Firmware P2 saved as a corrected firmware.

After Firmware P1 and Backup Firmware P2 are ensured all correct, both of the firmware should be compared if they are the same. After Firmware P1 is successfully renewed, Backup Firmware P2 is not yet backed up due to accidents possibly happen to the computer system, such as short circuit. Therefore, if the result is different after the comparability between Firmware P1 and Backup P2, Firmware P1 will be backup to Backup Firmware P2. If Firmware P1 and Backup P2 are the same after the comparability, Initial Program will be removed to Firmware P1, which can process the normal operation for computer system. Especially, this invention can be employed to necessity of the scanner.

With reference to Figure 2, the second portion as Firmware P1, the function of this portion for the computer system is really for controlling the computer system. The new firmware data will firstly be rewritten into this portion at any time when it is going to refresh the original firmware.

The third portion, as Backup Firmware (P2), this portion is for becoming as the backup file of the present firmware. Also, the main function of this portion is for reconstructing the data of Firmware P1 if Firmware P1 is wrong, as Figure 2.

The fourth portion, as P1 Firmware Parameter, this portion is for saving Firmware P1 and Backup Firmware P2 and their checksum value. The same with he fifth portion, as P2 Firmware Parameter, this portion is for saving Firmware P1 and Backup Firmware P2 and their checksum value. Also these portions will be provided for checking Firmware P1 and checking Backup Firmware P2 under Initial Program operation, as Figure 2.

After the above definition, some of possible conditions will happen in the preferred embodiment. Thus, the flow chart of this preferred embodiment will be processed under the following conditions and shown by Figure 3.

Figure 3 shows Condition1, which is under a normal execution without writing new firmware into the computer memory and starting from legend 20 of Figure 3:

(1) Checking Firmware P1, as legend 21 of Figure 3, it will be proper due to without writing operation;

(2) Checking Backup Firmware P2, as legend 22 of Figure 3, it will be proper due to without writing operation;

(3) Checking if Firmware P1 and Backup Firmware P2 is the same, as legend 23 of Figure 3, it will be the same due to without writing operation; and

(4) Back up to Firmware P1 and Firmware P1 will be executed if there is no error happened, as legend 24 of Figure 3.

Condition 2 is also indicates as Figure 3. Firmware P1 is written into the computer memory but the written operation is failed, so that the process is started from legend 20 of Figure 3:

(1) Checking if Firmware P1 is correct, as legend 21 of Figure 3, it could be wrong due to written operation is failed;

(2) Backup Firmware P2 is written into Firmware P1, as legend 25 of Figure 3. If it fails, it also can restart and go back to condition 3, as Figure 2, otherwise to be continued; and

(3) There is no error happened, Initial Program can be re-executed and it can go to Condition1, as Figure 2.

Figure 3 illustrates Condition 3 as well. A changeable firmware, when Backup Firmware P2 is successfully written into the computer memory but backup operation is failed so that the process is started from legend 20 of Figure 3:

(1)Checking if Firmware P1 is correct, as legend 21 of Figure 3, it could be right due to successfully writing;

(2)Checking if Backup Firmware P2 is correct, as legend 22 of Figure 3 it could be wrong due to backup operation is failed, it needs a backup operation;

(3)Firmware P1 will be written into Backup Firmware P2, as legend

26 of Figure 3. If it fails, it also can restart and go back to condition 4, as Figure 2, otherwise to be continued; and

(4)There is no error happened, Initial Program will be re-executed and back to condition1, as Figure 2.

Again, referring with Figure 3, Condition 4 for checking the purpose firmware is successfully written into the computer memory so that the process is started from legend 20 of Figure 3:

(1) Checking if Firmware P1 correct, as legend 21 of Figure 3, it is right due to a successful writing operation;

(2) Checking if Backup Firmware P2, as legend 22 of Figure 3, it is right without writing operation;

(3) Checking if Firmware P1 and Backup Firmware P2 are the same, as legend 23 of Figure 3. Firmware P1 and Backup Firmware P are different due to Firmware P1 is a changeable firmware but the Backup Firmware P2 is an original firmware, therefore Backup Firmware P2 will be backup;

(4) Firmware P1 is backup into Backup firmware P2, as legend 26 of Figure 3. It will go back to condition 4 if the backup operation is failed, as Figure 2. Then it can be continued if backup is successful; and

(5) There is no error happened, then Initial Program can be re-executed and back to condition1, as Figure 2.

According to this preferred embodiment, this invention can perform method steps for operating a firmware that concludes the steps of the following. Also, this invention can be used to the scanner.

Firstly a memory of the machine is divided as five portions which are able to provide space for storing a plurality of computer readable program. Sequentially an initial program can be installed into a first portion of the memory of the machine and as a computer readable fixed program. Then removing a first firmware from the memory of the machine would be carried out. A second firmware is

installed into a second portion of memory of the machine. A second firmware is backed up into a third portion of the memory of the machine. And finally, a plurality of parameter of the second firmware is installed into fourth portion of memory of the machine.

Finally, it is mentioned in the preferred embodiment, checking checksum of the firmware is the method for ensuring if the firmware is correct. Also, check checksum between any two of firmware will be the method for ensuring if both of the firmwares are the same. Therefore, in accordance with the present invention, a method is provided for protecting and upgrading firmware that substantially solves the above mentioned system error and possible faults to computer system. In addition, the computer system under operation still can recover the original firmware of original system, even though the errors happen under upgraded process. It can guarantee the system is under a normal condition. Also, it can set up the firmware under the safety condition. Of course, the parameter of all new five portions will be successfully renewed.

It is understood that various other modifications will be apparent to and can be readily made by those skilled in the art without departing from the scope and spirit of this invention. Accordingly, it is not intended that the scope of the claims appended hereto be limited to the description as set forth herein, but rather that the claims be construed as encompassing all the features of patentable novelty that reside in the present invention, including all features that would be treated as equivalents thereof by those skilled in the art to which this invention pertains.